Verification Diagrams

Up to now, we have presented the constituents of a proof by rules CHAIN, WELL, or DISTR-RANK by tables. An alternate presentation is provided by verification diagrams. A verification diagram is a directed graph such that:

- Nodes contain labeled assertions, identifying helpful situations.
- There exists a single node with no successors, called the terminal node, and labeled by the goal assertion q.
- Every node has a distinguished edge departing from it, and labeled by a transition which is helpful for this node. A node may have additional multiple unhelpful (indifferent) edges departing from it.

Diagrams differ by the rule they are supposed to represent.

Chain Diagrams

It is required that

- The terminal node is labeled by $h_0 : q$.
- If there is an edge connecting node h_i to node h_j, then $i > j$.

Assume that non-terminal node h_i has the helpful transition t_i which connects it to node h_j and the unhelpful successors $h_{k_1}, \ldots, h_{k_n}$. This implies the following verification conditions:

$$\begin{array}{llll} \text{C2.} & h_i \wedge \rho_t \Rightarrow & h'_i \vee h'_{k_1} \vee \cdots \vee h'_{k_n} & \text{For every } t \neq t_i \\ \text{C3.} & h_i \wedge \rho_{t_i} \Rightarrow & h'_j & \\ \text{C4.} & h_i \Rightarrow & En(t_i) & \end{array}$$

A CHAIN diagram is defined to be $\mathcal{D}$-valid if all the verification conditions associated with its nodes are $\mathcal{D}$-valid.

Claim 8. *If a verification diagram with nodes $h_0, \ldots, h_n$ is $\mathcal{D}$-valid then so is the temporal formula*

$$\bigvee_{i=0}^{n} h_i \quad \Rightarrow \quad \Diamond\, h_0$$

Corollary 9. *If, in addition, we establish the $\mathcal{D}$-validity of*

$$p \quad \Rightarrow \quad \bigvee_{i=0}^{n} h_i \qquad \text{and} \qquad h_0 \quad \Rightarrow \quad q$$

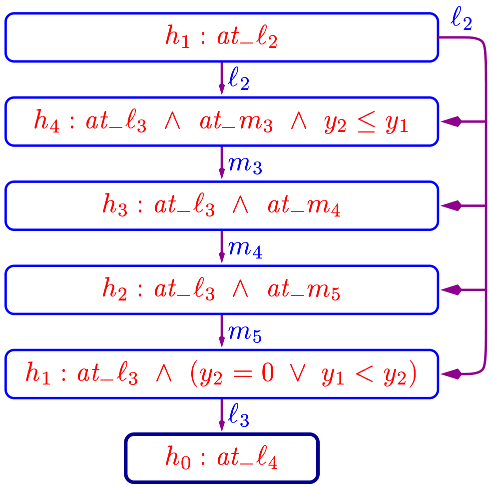
then we can conclude

$$p \quad \Rightarrow \quad \Diamond\, q$$

Example: BAKERY-2

$$
\textbf{local } y_1, y_2 : \textbf{natural initially } y_1 = y_2 = 0
$$

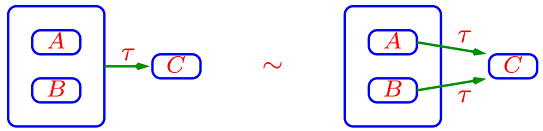
$$
P_1 :: \left[\begin{array}{l} \ell_0 : \textbf{loop forever do} \\ \left[\begin{array}{l} \ell_1 : \text{Non-Critical} \\ \ell_2 : y_1 := y_2 + 1 \\ \ell_3 : \textbf{await } y_2 = 0 \ \vee \ y_1 < y_2 \\ \ell_4 : \text{Critical} \\ \ell_5 : y_1 := 0 \end{array}\right] \end{array}\right] \parallel P_2 :: \left[\begin{array}{l} m_0 : \textbf{loop forever do} \\ \left[\begin{array}{l} m_1 : \text{Non-Critical} \\ m_2 : y_2 := y_1 + 1 \\ m_3 : \textbf{await } y_1 = 0 \ \vee \ y_2 \le y_1 \\ m_4 : \text{Critical} \\ m_5 : y_2 := 0 \end{array}\right] \end{array}\right]
$$



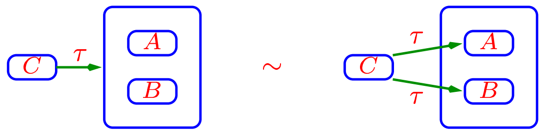
Encapsulation (Statecharts) Conventions

There are several conventions which make visual presentation more effective. We introduce compound nodes which may contains several internal nodes. The following graphical equivalences explain the conventions:

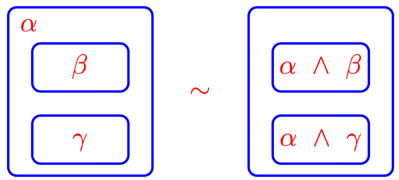
Departing Edges:



Arriving Edges:



Common Factors:

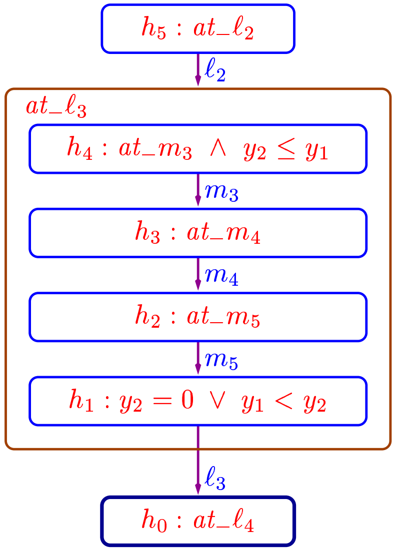


Encapsulated Verification Diagram for BAKERY-2

local y_1, y_2 : **natural initially** $y_1 = y_2 = 0$

$P_1 ::$
- ℓ_0 : **loop forever do**
  - ℓ_1 : **Non-Critical**
  - $\ell_2 : y_1 := y_2 + 1$
  - ℓ_3 : **await** $y_2 = 0 \;\vee\; y_1 < y_2$
  - ℓ_4 : **Critical**
  - $\ell_5 : y_1 := 0$

$\parallel$ $P_2 ::$
- m_0 : **loop forever do**
  - m_1 : **Non-Critical**
  - $m_2 : y_2 := y_1 + 1$
  - m_3 : **await** $y_1 = 0 \;\vee\; y_2 \le y_1$
  - m_4 : **Critical**
  - $m_5 : y_2 := 0$



WELL Diagrams

Node h_i contains also a ranking function δ_i. It is required that $\delta_0 = 0$.

Assume that non-terminal node h_i has the helpful transition t_i which connects it to node h_j and the unhelpful successors $h_{k_1}, \ldots, h_{k_n}$. This implies the following verification conditions:

$$\begin{array}{lll} \text{W2.} & h_i \wedge \rho_t \;\Rightarrow\; (h'_i \wedge \delta_i \succeq \delta'_i) \vee & \\ & \quad (h'_{k_1} \wedge \delta_i \succ \delta'_{k_1}) \;\vee\; \cdots \;\vee\; (h'_{k_n} \wedge \delta_i \succ \delta'_{k_n}) & \text{For every } t \neq t_i \\ \text{W3.} & h_i \wedge \rho_{t_i} \;\Rightarrow\; h'_j \wedge \delta_i \succ \delta'_j & \\ \text{W4.} & h_i \;\Rightarrow\; En(t_i) & \end{array}$$

A WELL diagram is defined to be $\mathcal{D}$-valid if all the verification conditions associated with its nodes are $\mathcal{D}$-valid.

Claim 10. *If a* WELL *verification diagram with nodes $h_0, \ldots, h_n$ is $\mathcal{D}$-valid then so is the temporal formula*

$$\bigvee_{i=0}^{n} h_i \quad\Rightarrow\quad \Diamond\, h_0$$

Corollary 11. *If, in addition, we establish the $\mathcal{D}$-validity of*

$$p \;\Rightarrow\; \bigvee_{i=0}^{n} h_i \qquad \text{and} \qquad h_0 \;\Rightarrow\; q$$

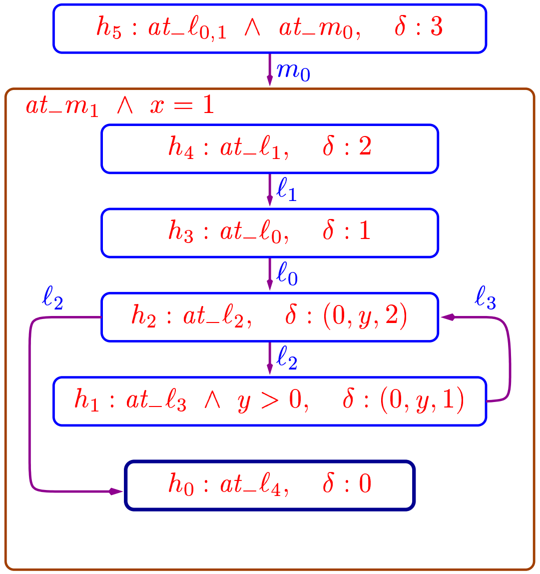
then we can conclude

$$p \;\Rightarrow\; \Diamond\, q$$

Apply to Program UP-DOWN

$x, \ y$: **natural initially** $x = y = 0$

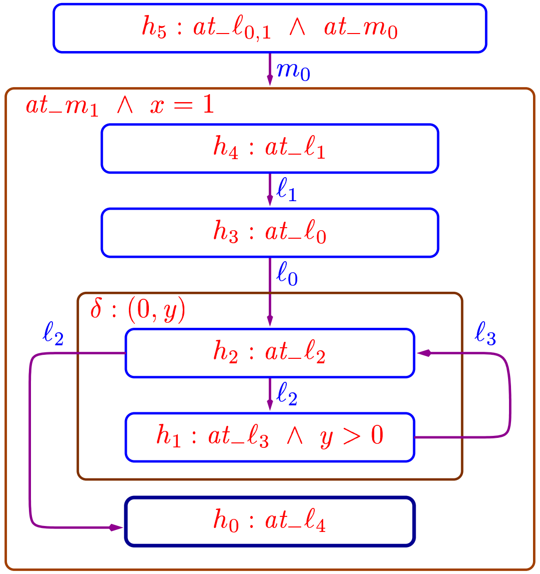
$$P_1 :: \left[\begin{array}{l} \ell_0: \ \textbf{while } x = 0 \textbf{ do} \\ \quad [\ell_1: \ y := y + 1] \\ \ell_2: \ \textbf{while } y > 0 \textbf{ do} \\ \quad [\ell_3: \ y := y - 1] \\ \ell_4: \end{array}\right] \quad \| \quad P_2 :: \left[\begin{array}{l} m_0: \ x := 1 \\ m_1: \end{array}\right]$$



Encapsulation Conventions Concerning Ranking

We adopt the additional conventions:

- In case node h_i does not have an explicit ranking labeling, it is as though it had the label $\delta : i$.
- In case a compound node has the transcription $\delta : f$ at its top left corner, the factor f is added as a left lexicographic component to all the rankings of the contained nodes.



δ_5 :	5
δ_4 :	4
δ_3 :	3
δ_2 :	$(0, y, 2)$
δ_1 :	$(0, y, 1)$
δ_0 :	0

Diagrams for Parameterized Systems

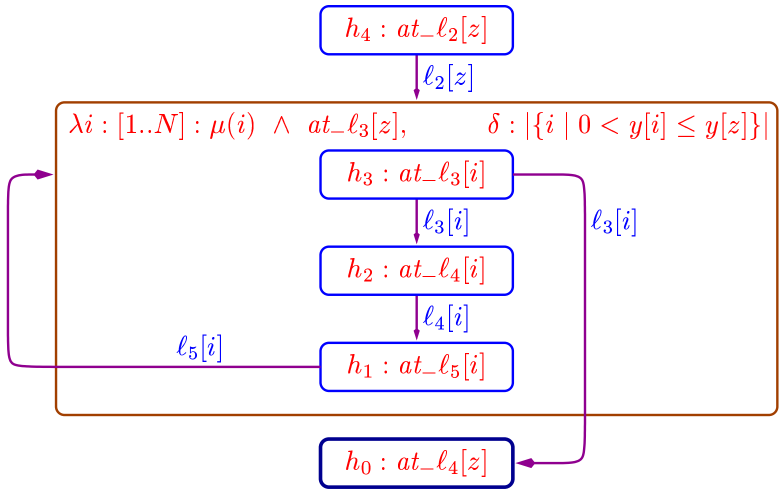
To deal with parameterized systems, we introduce the inscription $\lambda i : [1..N]$ labeling a compound node. This is equivalent to having N copies of this node, one for each value of $i \in [1..N]$. Assertions and transitions within the node may be parameterized by i.

Example: a Diagram for BAKERY

N : **natural where** $N > 0$

y : **array**$[1..N]$ **of natural where** $y = 0$

$$\overset{N}{\underset{i=1}{\|}} P[i] :: \left[\begin{array}{l} \ell_0\text{: } \textbf{loop forever do} \\ \quad \left[\begin{array}{ll} \ell_1\text{:} & \textbf{Non-critical} \\ \ell_2\text{:} & y[i] := \max(y[1], \ldots, y[N]) + 1 \\ \ell_3\text{:} & \textbf{await } \forall j \neq i : y[j] = 0 \ \vee \ y[i] < y[j] \\ \ell_4\text{:} & \textbf{Critical} \\ \ell_5\text{:} & y[i] := 0 \end{array}\right] \end{array}\right]$$



Apply to TOKEN-RING

local α : **array**$[1..N]$ **of boolean where** $\alpha[1] = 1, a[2] = \cdots = a[N] = 0$

$\overset{N}{\underset{i=1}{\parallel}} P[i] ::$

$S ::$
- ℓ_0 : **loop forever do**
  - ℓ_1 : **request** $\alpha[i]$
  - ℓ_2 : **if** $at\_m_2[i]$ **then**
    - $[\ell_3$: **await** $at\_m_4[i]]$
  - ℓ_4 : **release** $\alpha[i \oplus 1]$

$\parallel$

$C ::$
- m_0 : **loop forever do**
  - m_1 : **Non-critical**
  - m_2 : **await** $at\_\ell_3[i]$
  - m_3 : **Critical**
  - m_4 : **await** $\neg at\_\ell_3[i]$

Diagram for TOKEN-RING